TECHNICAL INFORMATION SYSTEM FOR COLLOBORATIVE MATERIAL RESEARCH

R. Chudoba*, C. Butenweg, W. Kuhlmann

Chair of Structural Statics & Dynamics, RWTH Aachen, Mies-van-der-Rohe-Str. 1, D-52056 Aachen, Germany

Abstract

This paper describes concepts used in a development of a technical information system for supporting collaborative material research. The system has been set up to support the development of textile-reinforced concrete (TRC) by applying the modern theoretical concepts of software and database engineering to integrate the available open source tools in an effective way. The system works as a database powered internet server with a transparent definition of the product and process model.

Keywords: product modeling, software engineering, database engineering, material modeling, model calibration.

1 Introduction

The goal of the collaborative research center (SFB 532) “Textile reinforced concrete: the basis for the development of a new material technology” installed in 1998 at the RWTH Aachen is a complex assessment of mechanical, chemical, economical and productional aspects of the developed material. The research project involves 10 institutes performing parallel investigations in 15 projects. The coordination of such a research process requires effective software support for information sharing in form of data exchange, data analysis and data archival. Flexible structuring of the data gathered from several sources is a crucial premise for a transparent accumulation of knowledge and, thus, for an efficient research in a long run.

The development of such a system requires models to describe and organize the information and activities involved in the overall research process. Examples of formal specification of the process and product models in the production process have been reported by Lee and Sause [1] and based on the entity-based approach.

The technical information system developed in the SFB 532 has been implemented as a database-powered web server with an object-oriented data model [2]. It serves as an intranet server with access domains devoted to the involved research groups. At the same time, it allows us to present the selected results just by granting a data object an access from the public area of the server via internet.

2 System design

2.1 System requirements

The first step in the software system design is the specification of the overall functionality provided by the system. The required functionality comprises the following tasks

- Design of experiments of various types with varying input and output parameters
- Dynamic construction of queries over the performed experiments
- Automatic accumulation of data related to the material components coming from processes involving experiments and simulations
- Automatic calibration of the simulation models
- Sensitivity analysis with respect to selected input parameters
- Validation of the developed simulation models

Except of the functional requirements we have imposed the following technical requirements

- The TIS must be accessible through standard HTML-based browsers
- Simple integration of external simulation packages

2.2 Architecture

In order to fulfill the technical requirements we have chosen the so called LAMP architecture comprising the Linux, Apache, MySQL [3] and PHP [4] to set up a database powered database server [LAMP]. Instead of applying an object-oriented database an relation database (MySQL) is used and provided it with object-oriented features via “object-relational mapping”, which is implemented in the wrapping library called HODBASE (Figure 1). The reason for this choice was the intention to make the data representation independent on a particular database product or vendor. Thus, we could apply any database engine based on the standard structured query language interface without data conversions and code modifications.

The HODBASE library is implemented with an application programming interface (API) in the PHP language and Python language [5] to support the dynamic generation of HTML pages based on the incoming queries. The Python API allows to integrate external simulation and data analysis packages into the system. Using this API the finite element simulation toolkit ORFEUS was integrated into the TIS. In order to enable direct connection to standard office programs for standardized input and output tasks the ODBC interface is used.

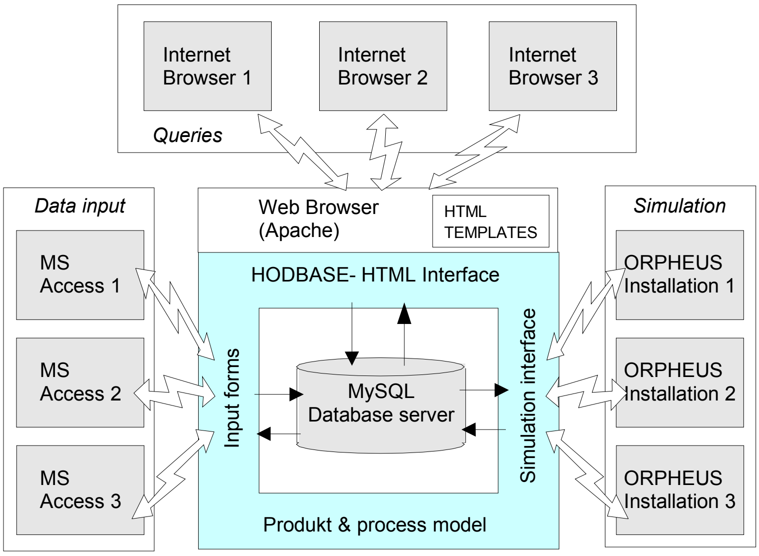


Figure 1: System architecture

2.3 Database server with object-relational mapping

In order to model the products and processes involved in the material research we need a semantically rich data model with the concepts of object identity, object classes, inheritance and access control. The applied object-relational mapping provides an access to database objects based on the "HTTP query address" specification. For example, accessing an object in the class ExSequence with a unique ID 234 and performing an action edit is specified as follows:

http://sfb532.rwth-aachen.de/index.php?class=ExSequence&SID=234&action=edit

The HODBASE library contains the database scheme and can translate the incoming query address into SQL queries over several tables:

```
SELECT   ExSequence.*,ExSequence.class
FROM     ExSequence
WHERE    (ExSequence.class='ExSeqComposite') and ExSequence.SID=234
```

The result of the query is formatted into a new HTML page and subsequently sent back to the browser. The links in the generated page specify further "HTTP query addresses" navigating to other objects in the database.

The object-relational mapping involves not only individual objects but also the object attributes. Complex attributes and container attributes are translated into additional queries as well. Using attribute mapping we can represent the relationships of the type one-to-one, one-to-many and many-to-many. Further attribute types allow us to include data with special behavior during store and retrieval. For example files, images and passwords are realized by corresponding specialized attribute types.

2.3 Access control

The access to the system is controlled by an authentication subsystem which reflects associates each user with a set of access permissions. During the retrieval of the data it is checked whether the user has the permissions required by the class, by the object and by the attribute to be retrieved. Further, the permissions are checked for each action to be invoked. In this way, it is possible to define data regions to be accessible or modifiable by different groups of users (Figure 2).

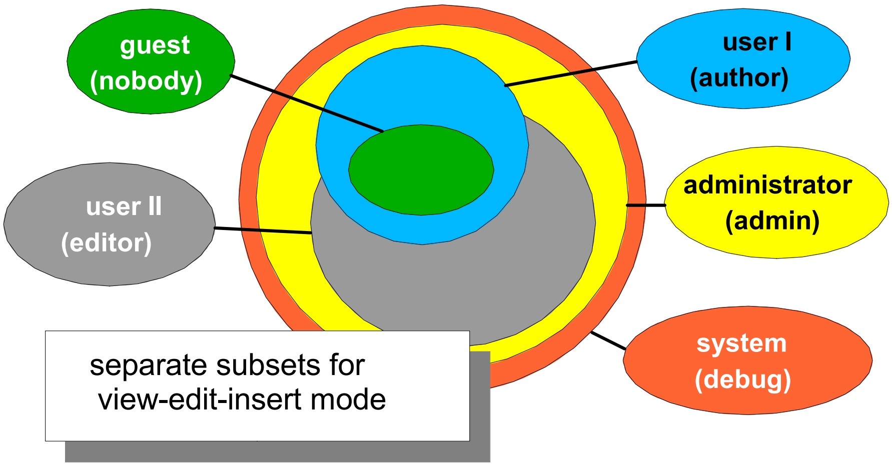
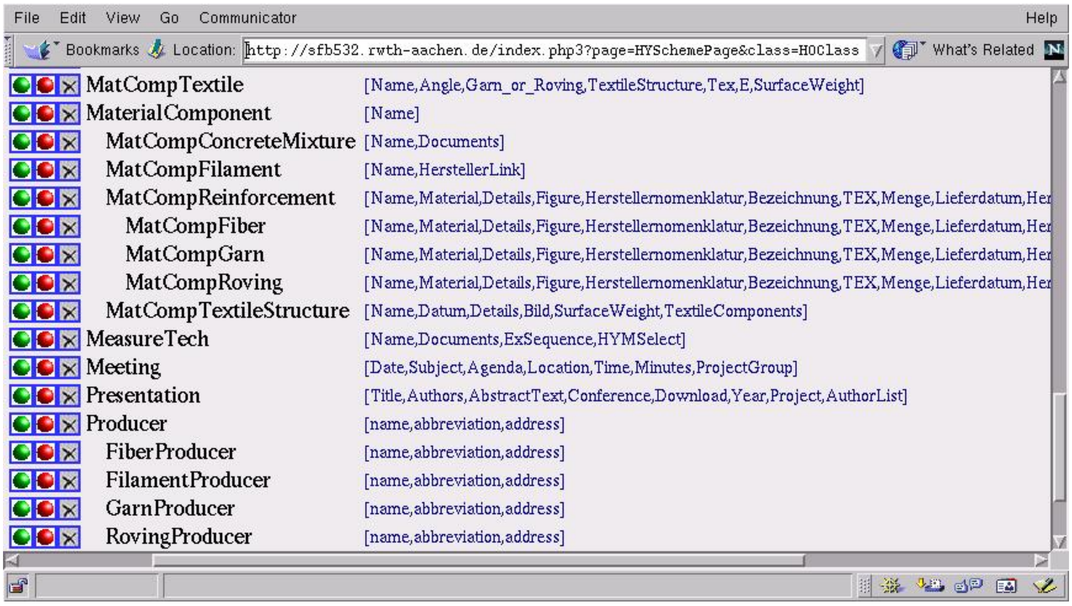


Figure 2: Access regions

2.4 Administration

The class definitions are stored in the database and may be manipulated as any other objects. The addition or deletion of attributes is performed though the HTML interface. Whenever a class definition gets changed, the corresponding object-oriented interface code in PHP and Python scripting language gets generated and the relational database scheme gets updated though SQL. The snapshot in the Figure 3 shows the class definitions in the HTML interface.



Location: http://sfb532.rwth-aachen.de/index.php3?page=HYSchemePage&class=HOClass

MatCompTextile	[Name,Angle,Garn_or_Roving,TextileStructure,Tex,E,SurfaceWeight]
MaterialComponent	[Name]
MatCompConcreteMixture	[Name,Documents]
MatCompFilament	[Name,HerstellerLink]
MatCompReinforcement	[Name,Material,Details,Figure,Herstellernomenklatur,Bezeichnung,TEX,Menge,Lieferdatum,Her
MatCompFiber	[Name,Material,Details,Figure,Herstellernomenklatur,Bezeichnung,TEX,Menge,Lieferdatum,Her
MatCompGarn	[Name,Material,Details,Figure,Herstellernomenklatur,Bezeichnung,TEX,Menge,Lieferdatum,Her
MatCompRoving	[Name,Material,Details,Figure,Herstellernomenklatur,Bezeichnung,TEX,Menge,Lieferdatum,Her
MatCompTextileStructure	[Name,Datum,Details,Bild,SurfaceWeight,TextileComponents]
MeasureTech	[Name,Documents,ExSequence,HYMSelect]
Meeting	[Date,Subject,Agenda,Location,Time,Minutes,ProjectGroup]
Presentation	[Title,Authors,AbstractText,Conference,Download,Year,Project,AuthorList]
Producer	[name,abbreviation,address]
FiberProducer	[name,abbreviation,address]
FilamentProducer	[name,abbreviation,address]
GarnProducer	[name,abbreviation,address]
RovingProducer	[name,abbreviation,address]

Figure 3: Class definitions

3 Product model

The classes defining the object-oriented database scheme is decomposed in two categories: the experimental classes and the material component classes. The **experimental classes** serve to describe particular experiments and experiment types to be used in a particular research task. The **material component classes** are used to describe the material components themselves with a characteristic behavior which is independent on the use in a particular experiment or product.

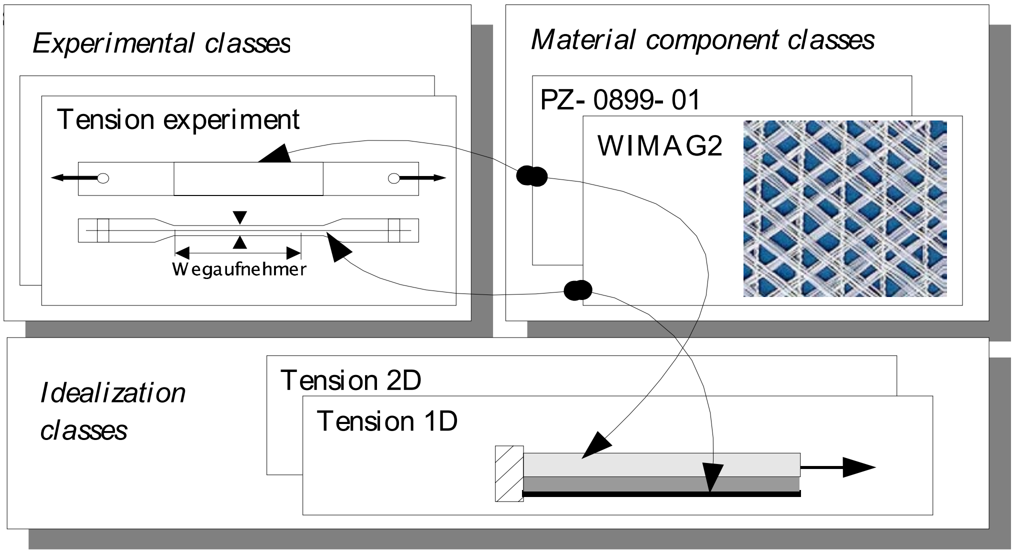


Figure 3: Class categories

3.1 Classes describing experiments

More specifically, experimental classes describe products with dimensions and are composed of particular material components. Furthermore, they are associated with affects in form of input and output parameters imposed for the experimental or numerical use.

The general definition of an experimental setup is concentrated in the class ExperimentType. This class defines the possible input and output parameters of an experiment and the possible composition of material components. This class serves as a template for designing an experiment sequences (class ExSequence) to explore a particular response using the experiment type, for example sensitivity with respect to selected input parameters or different choices of material components. The specification of the input and output parameters leads to an automatic generation of a class for the individual experiment runs. This may be expressed in pseudo C++ notation as follows:

```
structExSample< class ExSequence >
  ExSequence::input_p    input_params;
  ExSequence::output_p  output_params;
};
```

This demonstrates the way how users can extend the database scheme in a controlled way. The HTML user interface provides object methods that can modify database scheme for selected classes. These methods are essentially instantiation methods for generic template classes predefined in the class diagram of the database.

3.2 Classes describing material components

The material component classes are used to construct material combinations applied in experiments and simulations and to accumulate all the data gathered for the particular material components. Examples of such data are material parameters characterizing the material behavior independently on the dimensions and boundary conditions of a particular experiment sample.

The material components distinguished in the textile reinforced concrete are the following are shown in the class diagram in Figure The filaments are decomposed into the raw filament material and filament surface because these possess material characteristics of their own (coating, surface roughness, contact angle). Filament assemblies are called fibers and may be realized either as rovings or yarns. Assemblies of fibers are two and three dimensional textile structures.

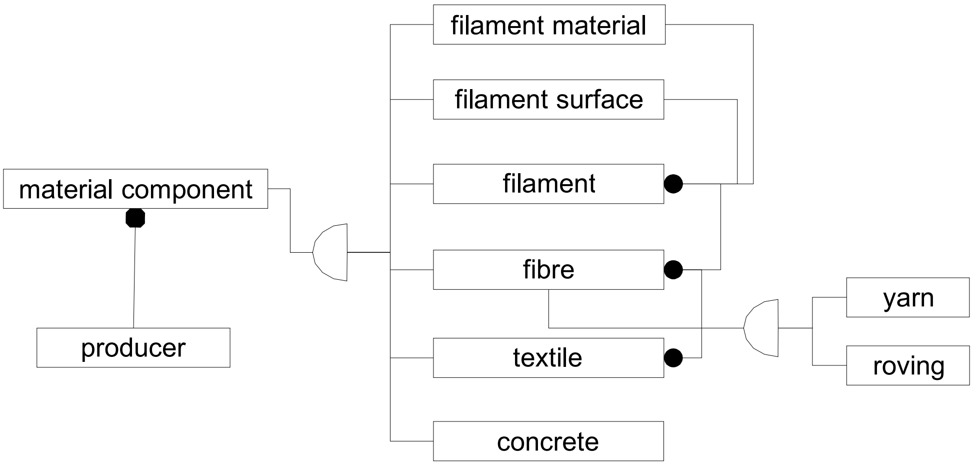


Figure 4: Class diagram for material components

3.3 Classes defining idealizations

The material parameters are quantified with respect to an idealization of the material behavior in a material model applied in the model of the experimental set up. Depending on the complexity of the material model the type and number of material parameters to be associated with material components can vary. There are several possible idealizations which can be associated with a particular experiment type.

Idealization classes are associated with a particular experiment type. They are constructed in connection with a particular experiment sample and can extract the experimental input and output data needed for the computation. Further, the material components are retrieved. The connection to the material component allows the retrieval of the material parameters required in the numerical simulation.

The idealization objects can be manipulated through the ORFEUS simulation toolkit. The data transfer between the database and the ORFEUS toolkit is performed through the Python scripting interface of the database. The scripting interface allows us an easy integration of external simulation tools into the TIS. For this purpose the application of the wrapping technique which allows us to equip external software with a scripting interface is extremely helpful [6].

3 Process model

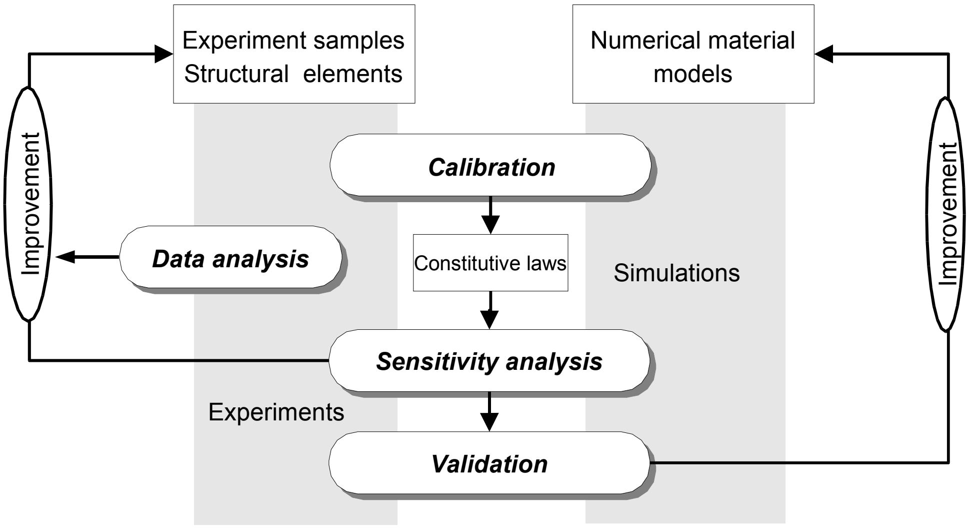


Figure 5: Overall development process model.

The structures defined in the product model are manipulated within the "use cases" involved in the overall process of material research and development. The iterative nature of the process encompasses identification of parameters of a particular material model (calibration), sensitivity analysis and parametric studies with respect to selected input parameters and validation of the developed material models to assess their quality. These "use cases" are defined as classes within the TIS with a state diagram defining the procedure to assisting in the particular research task.

The mentioned processes are represented as templates which are instantiated by the system administrator to an input procedure in one of the user interfaces (see Figure 1). The templates are conceptual and can be realized through

- the browser HTML/PHP interface for queries and navigation,
- the ODBC interface for bulk data input via standard office programs,
- the scripting Python interface to control the computation of the simulation toolkit.

3.1 Experiment design

The systematic archival of the experimental data within the previously described product model is carried out by using the following procedure:

1.Create/select an experiment type
2.Specify the goal of the experiment
3.Select input parameters to be varied in the experiment sequence
4.Select output parameters measured in the experiment sequence
5.Fill in the input and output data for each experiment sample

This procedure allows us transparent archival and logging of experimental data. The explicit specification of input and output parameters makes it possible to keep track of the experiments and their motivations performed in the past. This specification is performed within the HTML interface. After setting up the experiment, the input and output data is inserted through an ODBC based interface realized via predefined input forms in a standard office application program (e.g. MS Access).

3.3 Calibration of material models

The quantitative characteristics of material components like material parameters, constitutive laws, statistical characteristics of the material structure are quantified by using idealizations. The material parameters are identified either in a closed form, e.g. Young modulus, or in an iterative form by fitting the response of the idealized model to the experimental response. Thus, the product model reflects the association between an experiment sample or group of samples with the same material composition with an idealization object to identify the material parameters.

The calibration procedure controlled by the TIS is shown in the Figure 5. The first step is the selection of the experiment sample(s) and the idealization with the material model to calibrate. The general procedure of calibration is implemented by a process template. Prior to the start of calibration the process template must perform two initial steps: (1) identify the compatible response values of the experiment sample and the idealization (2) localize the slots of material parameters in the material components occurring in the current material model. The classes representing the response must be comparable and integral measures must be defined to quantify the lack of fit between the experiment and simulation.

In the Figure 5 calibration procedure is applied to the bond material model calibrated by pull-out of a filament from the concrete matrix. The iteration has been controlled by the genetic algorithm and allowed us an automatic determination of the 5 material parameters of the material model in 500 simulations.

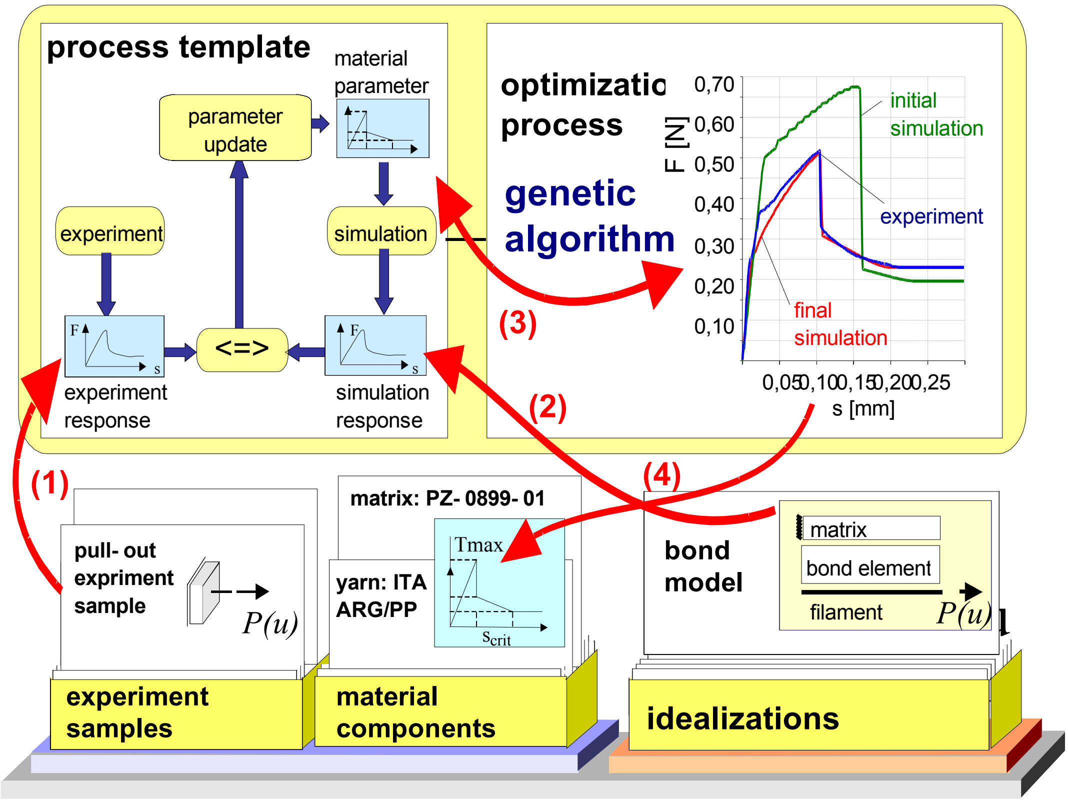


Figure 5: Calibration of bond material model

3.4 Sensitivity analysis and parametric studies

There are two types of sensitivity analysis distinguished by the system, the purely experimental method of “design and analysis of experiments” based on generalized regression model [7, 8] and the sensitivity analysis supported by an idealization based on mechanical and physical theory.

In the former case, there is a number of software packages applying regression models to evaluate the significance and interactions of the varied input parameters. The definition of experimental classes described previously has been developed with regard to the methodology of design of experiments. This makes it possible to provide an interface between the database and external DOE tools like DesignExpert [9].

In case that robust material models are available, the sensitivity analysis may be supported by numerical analysis. An important issue in this case is the knowledge of the quality of the material model at hand which can only be obtained through the systematic validation [10]. The definition of the process templates for validation of models and sensitivity analysis is the subject of the further development.

4 Conclusions

The complex material research involves many aspects to be explored simultaneously at various levels of research. An effective information sharing and exchange can essentially increase the efficiency of research and enable more targeted progress of material development. The present article summarized the applied concepts of software engineering that have been applied in the system design of the TIS for textile reinforced concrete. The major goals of the TIS is to support (1) the experimental design in a multi user environment, (2) the data mining and data analysis, (3) the calibration of newly developed material models, (4) the sensitivity analysis with respect to input parameters and (5) the validation of the developed material models.

Acknowledgement

The present work has been carried out in the framework of the project “Technical information system for the development of textile reinforced concrete” included in the collaborative research center “Textile reinforced concrete: foundation of a new technology” (SFB 532) sponsored by the German research foundation.

References

[1] C. Lee, R. Sause, N. Hong, *"Overview of entity-based integrated design product and process models"*, Advances in Engineering Software, 29(10), 809-823, 1998.

[2] R. Chudoba, W. Kuhlmann, *“Ein technisches Informationssystem für Controlling und Präsentation des Sonderforschungsbereiches 532”*, 1. Fachkolloquium der Sonderforschungsbereiche 528 und 532, Aachen, 2001.

[3] MySQL relational database, www.mysql.org.

[4] PHP programming language, www.php.net.

[5] Python programming language, www.python.org.

[6] Object-relational mapping, www.object-relational.com.

[7] R. Chudoba, Z. Bittnar, Z. Littwin, C. Könke, *"Rapid prototyping of performance assessing algorithms"*, ACTA Polytechnica, 99(2), 129-141, 1999.

[8] D. Montgomery, *"Design and Analysis of Experiments"*, John Wiley & Sons, 2001.

[9] A. Dean, D. Voss, *"Design and Analysis of Experiments"*, Springer, 1999

[10] J.P.C. Kleijnen, R.G. Sargent, *"A methodology for fitting and validating metamodels in simulation"*, European Journal of Operational Research, 82, 145-162, 2000.

[11] Design-Expert 6 for Windows, Software for Design of Experiments, www.statease.com.